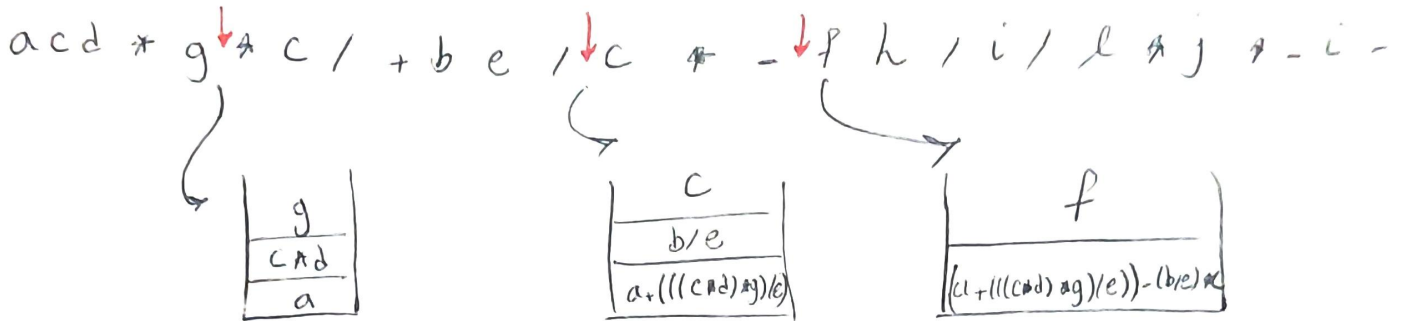


(۱)

$$- i - j + (f / h) / i - ((b / e) * c) - (a + ((c * d) * g) / e)$$



(۲)

به لحاظ اینکه همان علامه و نحوه Top و Min را نیز در متن زیر می بینیم. بدین ترتیب که مقدار اولیه ی

متغیر Min به ∞ (Null است) (create)

۱۱. به هنگام push، وقتی هر عدد به stack می شود باید مقدار آن item با مقدار فعلی Min مقایسه شود.

از جمله item در Min کمتر بود یا $Min == Null$ بود آنگاه جمله Min با item مساوی قرار داده شود.

- به هنگام push اگر صفی - stack اضافه شود که در Min فعلی کمتر است - جای آن عدد - عددی با خرابی

- بسته اضافه می شود تا - به هنگام pop نحوه خروج Min شود و باید به تقسیم ضرب Min جدید را پیدا کنیم.

در صورتی که عدد جدیدی که قصد افزودن آن را داریم از Min فعلی کمتر باشد - بسته مقدار زیر را اضافه می کنیم:

$$Top\ value = Min - Min \times 2$$

۱۲. به هنگام pop اگر عددی که pop می شود از Min بزرگتر بود خود عدد بزرگتر از Min را اگر عددی در

Min کمتر بود پس به اینی Min تغییر کرده پس حالا باید Min فعلی را حساب کنیم که از عددی بزرگ

Top value، Min بزرگتر است که حاصل عدد - Min قدیم و Min حال حاضر

بسته را بزرگتر می کنیم!

(3)

یک متغیر را بعنوان نمادگر Node انتخاب می‌کنیم و با head متغیر مشخصی اوست می‌کنیم (myNode). همچنین متغیر
 شماره مقدار int را نیز برابر 1 قرار می‌دهیم! (توجه داشته باشید که اول اینها را می‌نویسیم) حال به خاتمه می‌رسیم
 و C++ انجام می‌دهیم. ~~در اینجا می‌توانیم از یک متغیر دیگر استفاده کنیم~~ از این آنگاه یک متغیر افشان
 افشان می‌کنیم اگر عددی برای ~~در اینجا~~ که افشان آن $\frac{1}{n}$ است myNode را برابر با آن Node انتخاب
 شده قرار می‌دهیم. در صورتی که می‌خواهیم myNode ننویسیم باید هیچ خالی دیگری انتخاب نمود که افشان آن برای n می
 $\frac{1}{n}$ است. (کوتاه می‌نویسیم، برای اولی 1، برای دومی $\frac{1}{m}$ و ... برای n امی $\frac{1}{n}$ است) ^{uniforms}
 به عبارت دیگر $\frac{1}{n} + \frac{1}{n} + \dots + \frac{1}{n} = 1$ است که حاصل ضرب افشان با مقدار $\frac{1}{n}$ برابر است در حال شده
 هرگاه Node انتخاب شده و در myNode قرار گیرد آن را به یک linked list اضافه می‌کنیم و در نهایت
 شده پس نیز k بار شدن ~~حل~~ را نمود چرا که برای انتخاب Node k ام نیاز به k بار شدن است
 در نهایت باید linked list را به ابتدای linked list ارجع اضافه کرد یا در واقع

$$\text{lastNode.next} = \text{head}$$

(4)

روش به صورتی می‌نویسیم که شیء اصلی ما شیء S_1 و شیء دیگری ما شیء S_2 است!

- 1) enqueue: برای این امر تنها کافی است که عناصر جدیدمان را در شیء S_1 که اصلی است push کنیم
- 2) dequeue: ابتدا تمامی عناصر S_1 را به جز آخری (توقتی Top) از S_1 اصلی pop کرده و به S_2 (ملکی) push می‌کنیم. سپس تنها عنصر باقی مانده در S_1 را pop کرده و به S_2 می‌نویسیم. حال
 ترتیب عناصر در S_2 برعکس است از بهر این چنین معنی باشد می‌توانیم است در این صورت عناصر عناصر S_1
 را ~~در S_2 pop~~ از S_2 pop و به S_1 push می‌کنیم!

⑤ حال حاضر شده اند یک بسته به نام S و یک آرایه به نام R استفاده می کنیم.

۱۱. ابتدا آرایه مقدار ۱- را append می کنیم.

سپس آخرین عضو آرایه را به عنوان (A) را در S می کنیم (ATmax) به S، push می کند

۱۲. در گام دوم عضو بعدی از آرایه را به بسته اضافه می کنیم | اگر $S.pop \leftarrow S.pop$ و آرایه ای از بسته اضافه می کنیم

اگر $S.pop \leftarrow S.pop$ مقدار آرایه بسته را خارج کرده

و به مرحله ۲ باز می گردیم و دوباره هر دو شرط را بررسی می کنیم تا زمانی که عضو آخر بسته برسم که در عدد در آرایه
نبردند و یا برابر ~~(-1)~~ باشند! و آن عدد را pop می کنیم (یا بزرگتر از عدد و یا آخرین عضو بسته)
و به ابتدای آرایه R اضافه می کنیم.

S: stack R: array A: input-array

شبهه

~~for (i = A.size() - 1; i >= 0; i--)~~

for (i = A.size() - 1; i >= 0; i--)

S.push(-1)

~~R.append(-1)~~

S.push(A[0])

while (S.pop < A[i]) ~~||~~ S.Top != 1

S.pop

R.appendFirst(S.pop)

S.push(A[i])

}

appendFirst is a function to add data at the start of array.